

## PENGENALAN POLA TEKSTUR BRODATZ DENGAN METODE JARAK *EUCLIDEAN*

ZA'IMATUN NISWATI

Program Studi Teknik Informatika, Fakultas Teknik, Matematika dan IPA  
Universitas Indraprasta PGRI  
[zaimatunnis@gmail.com](mailto:zaimatunnis@gmail.com)

**Abstract.** This research has a purpose to create a program aids Brodatz texture recognition using Euclidean distance-based Matlab so by inserting a new object (image texture pattern that had never stored in the file), it can be determined that the texture pattern similar to the pattern of the test texture. Euclidean Distance can be used in a statistical approach for texture feature extraction, to calculate the value of contrast, homogeneity, entropy, energy and the correlation. Matlab is a programming application that has been known for manufacturing applications supporting research. Matlab is a programming application in digital image processing that provides a variety of tools that will shorten the time of writing the program so that researchers focus more on results and innovation research. The result is the use of Euclidean Distance for texture pattern recognition is quite good and depending on the color intensity values, on average, entropy, energy, and homogeneity.

**Kata kunci :** texture recognition, Euclidean Distance, Brodatz texture, Matlab

### PENDAHULUAN

Dalam beberapa bidang seperti perdagangan, pemerintahan, pendidikan, kesehatan, dan pencegahan kriminal telah digunakan citra digital dalam jumlah yang besar. Kebutuhan informasi dalam bentuk citra sangat berkembang pesat seiring dengan bertambahnya koleksi citra yang ada di dalam database yang berskala besar.

Teknologi informasi yang kian berkembang dewasa ini telah banyak menghasilkan berbagai aplikasi yang menggunakan citra sebagai sumber informasi. Hal ini dikarenakan secara umum sebuah citra dapat memberikan informasi khusus yang berkaitan dengan identifikasi berbasis pengenalan ciri yang dapat dimanfaatkan dalam suatu sistem elektronik.

Ciri merupakan suatu tanda yang khas, yang membedakan antara satu citra dengan citra yang lain. Pada dasarnya suatu citra memiliki ciri-ciri dasar yaitu warna, tekstur, dan bentuk. Akan tetapi, ciri tekstur adalah hal yang paling penting dalam visi komputer karena banyak citra natural yang dapat dilihat sebagai komposisi dari tekstur yang berbeda (Kokare, 2007). Tekstur adalah karakteristik yang penting untuk analisis ciri berbagai jenis citra. Sehingga dalam permasalahan ini, hanya konsentrasi pada masalah menemukan fitur tekstur yang mirip dengan metode jarak *Euclidean*. Fitur ciri yang digunakan adalah fitur tekstur. Setelah diperoleh vektor fitur, langkah selanjutnya adalah menghitung jarak antara citra *query* dengan citra dalam database. Kemudian, nilai jarak tersebut diurutkan untuk mengetahui citra yang mempunyai kemiripan dengan citra *query*.

Tujuan penelitian ini adalah untuk membuat program bantu pengenalan tekstur Brodatz menggunakan metode jarak *Euclidean* berbasis Matlab sehingga dapat diketahui tekstur mana yang paling mirip dengan tekstur uji.

## TINJAUAN PUSTAKA

### 1. Tekstur

Walaupun belum ada definisi secara formal tentang tekstur, secara intuitif tekstur menyatakan ciri dari permukaan objek yang menggambarkan pola visual. Ciri ini berisi informasi tentang komposisi struktur permukaan, seperti misalnya awan, daun, batu bata dan kain. Selain itu juga menjelaskan hubungan antara permukaan untuk lingkungan sekitarnya (Al-Tayeche, 2003).

Sehingga tekstur menjadi salah satu fitur yang penting. Ciri tekstur antara lain meliputi kehalusan (*smoothness*), kekasaran (*coarseness*), dan keteraturan (*regularity*). Penggunaan fitur tekstur telah banyak digunakan secara luas oleh peneliti dalam menyelesaikan masalah pengenalan pola (*pattern recognition*) dan *computer vision*. Secara umum, representasi tekstur dapat diklasifikasikan menjadi dua, yaitu : struktural dan statistik (Gonzales, et al, 2002).

Untuk representasi tekstur secara statistic, dilakukan dengan menganalisa distribusi statistic dari intensitas citra termasuk diantaranya : *Fourier power spectra*, *co-occurrence matrices*, *shift-invariant principal component analysis* (SPCA), *Tamara feature* (terdiri dari *coarseness*, *contrast*, *directionality*, *linelikeness*, *regularity*, dan *roughness*), *Wold decomposition* (terdiri atas *harmonic*, *evanescent*, dan *indeterministic*), *Markov random field*, *fractal model*, dan teknik *multi-resolution filtering* seperti *Gabor* dan *wavelet transform*. Selain itu fitur tekstur (Gonzales and Wintz, 2004) terdiri atas *mean*, *standard deviation*, *third moment* dan *smoothness*. Tekstur yang diajukan Haralick (Haralick, 1998) terdiri atas *maximum probability*, *element-difference moment*, *inverse element-difference moment*, *uniformity* dan *entropy*.

Tekstur adalah konsep intuitif yang mendeskripsikan tentang sifat kehalusan, kekasaran, dan keteraturan dalam suatu daerah/wilayah (*region*). Dalam pengolahan citra digital, tekstur didefinisikan sebagai distribusi spasial dari derajat keabuan di dalam sekumpulan piksel yang bertetangga.

Secara umum tekstur mengacu pada pengulangan elemen-elemen tekstur dasar yang disebut primitif atau teksel (*texture element-textel*). Syarat-syarat terbentuknya suatu tekstur antara lain :

1. Adanya pola-pola primitif yang terdiri dari satu piksel atau lebih. Bentuk-bentuk pola primitif ini dapat berupa titik, garis lurus, garis lengkung, luasan, dan lainlain yang merupakan elemen dasar dari sebuah tekstur.
2. Pola-pola primitif tersebut muncul berulang-ulang dengan interval dan arah tertentu sehingga dapat diprediksi atau ditemukan karakteristik pengulangannya.

Suatu citra memberikan interpretasi tekstur yang berbeda apabila dilihat dengan jarak dan sudut yang berbeda. Manusia memandang tekstur berdasarkan deskripsi yang bersifat acak, seperti halus, kasar, teratur, tidak teratur, dan sebagainya.

Hal ini merupakan deskripsi yang tidak tepat dan non-kuantitatif, sehingga diperlukan adanya suatu deskripsi yang kuantitatif (matematis) untuk memudahkan analisis, (Falasev, RS, 2010)

### 2. Analisis Tekstur

Dalam pengolahan citra terdapat analisis citra. Analisis citra bertujuan mengidentifikasi parameter-parameter yang diasosiasikan dengan ciri dari objek dalam citra yang kemudian digunakan untuk menginterpretasikan citra. Analisis citra terdiri dari tiga tahapan yaitu ekstraksi ciri, segmentasi dan klasifikasi. Salah satu cara yang sering digunakan dalam memilah-milah citra adalah segmentasi, yaitu membagi citra menjadi bagian yang diharapkan termasuk objek-objek yang dianalisis. Segmentasi sering dideskripsikan sebagai proses analogi terhadap proses pemisahan latar depan dan latar

belakang. Tekstur merupakan karakteristik intrinsik dari suatu citra yang terkait dengan tingkat kekasaran (*roughness*), granularitas (*granulation*), dan keteraturan (*regularity*) susunan struktural piksel. Aspek tekstural dari sebuah citra dapat dimanfaatkan sebagai dasar dari segmentasi, klasifikasi, maupun interpretasi citra. Tekstur didefinisikan sebagai fungsi dari variasi spasial intensitas piksel (nilai keabuan). Berdasarkan strukturnya, tekstur dapat diklasifikasikan dalam 2 golongan (Karyanti, Yuli, 2010), yaitu makrostruktur dan mikrostruktur.

Analisis tekstur merupakan dasar dari berbagai macam aplikasi, aplikasi dari analisis tekstur antara lain : penginderaan jarak jauh, pencitraan medis, identifikasi kualitas suatu bahan (kayu, kulit, tekstil, dan lain-lain), dan juga berbagai macam aplikasi lainnya. Pada analisis citra, pengukuran tekstur dikategorikan menjadi lima kategori utama yaitu : statistis, struktural, geometri, model dasar, dan pengolahan sinyal. Pendekatan statistis mempertimbangkan bahwa intensitas dibangkitkan oleh medan acak dua dimensi (Jackson, 2009)

### 3. Pengenalan dengan Perhitungan Jarak *Euclidean*

Sebuah objek mempunyai banyak variasi pola yang dapat dijadikan dasar informasi untuk mengenali objek tersebut. Misalnya, bentuk tekstur bisa mempunyai banyak variasi penampakan yang diperoleh dengan melakukan, variasi sudut pandang, maupun variasi pencahayaan.

Proses pengenalan yang terjadi pada suatu sistem pengenalan pola pada umumnya adalah dengan membandingkan suatu pola masukan dengan pola yang telah tersimpan pada sistem tersebut. Setelah melakukan proses pelatihan, maka akan dihasilkan suatu pola latih yang pada metode ini berupa vektor ciri yang berisi komponen utama dari sejumlah citra latih (citra yang dilatihkan ke sistem). Sejumlah vektor ciri tersebut disimpan dalam suatu matrik, dan akan dikeluarkan pada saat proses pengenalan. Untuk proses pengenalan, sistem untuk proses pengenalan yang memiliki dimensi yang sama dengan citra latih telah disajikan ke sistem. Citra uji tersebut kemudian diekstraksi ciri dengan cara mengalikan dengan vektor eigen citra latih, dan akan menghasilkan vektor ciri berisikan komponen utama yang memiliki dimensi yang sama dengan vektor ciri citra latih.

Setelah didapatkan vektor ciri dari citra uji, maka proses selanjutnya adalah membandingkan vektor ciri dari citra uji dengan vektor ciri citra latih. Perbandingan tersebut dapat dilakukan dengan cara menghitung jarak euclidean (*euclidean distance*), yang merupakan selisih nilai piksel antara 2 vektor tersebut. Jarak *euclidean* adalah akar dari matematis dapat dirumuskan:

$$\text{Dist}(i,k) = \sqrt{\sum_{i=j}^D (i_j - k_j)^2}$$

Dengan  $\text{dist}(i,k)$  adalah jarak euclidean antara vektor  $i$  dan vektor  $k$

$i_j$  = komponen ke  $j$  dari vektor  $i$

$k_j$  = komponen ke  $j$  dari vektor  $k$

$D$  adalah jumlah komponen pada vektor  $i$  dan vektor  $k$ .

Dari hasil perhitungan jarak *euclidean* tersebut dapat ditentukan suatu citra tekstur adalah mirip bila memiliki jarak yang paling dekat atau nilainya paling kecil (AGani, 2012).

Sebuah citra mempunyai beberapa ciri yang digunakan untuk mengenali citra tersebut, antara lain:

- Intensitas warna ( $\sigma$ )
- Nilai rata-rata ( $\mu$ )
- Entropi ( $e$ )
- Energi ( $E$ )
- Homogeinitas ( $H$ )
- Contrast ( $C$ )

$$\text{Standar deviasi } \sigma = \sqrt{\frac{1}{N} \sum_{i=1}^N (x_i - \bar{x})^2}$$

$$\text{Nilai rata-rata} = \mu = \frac{1}{N} \sum_{i=1}^N x_i$$

$$\text{Entropi} = e = \sum_{i=1}^N p(x_i) \log P(x_i)$$

$$\text{Energi} = E_j = \frac{1}{M \times N} \sum_{x=1}^M \sum_{y=1}^N [P_j(x, y)]^2$$

$$\text{Homogeinitas} = H = \sum_i \sum_j \frac{P_d(i, j)}{1 + |i - j|}$$

$$\text{Contrast} = c = \sum_i \sum_j (i - j)^2 P_d(i, j)$$

#### 4. Pendekatan Statistik untuk Ekstraksi Fitur Tekstur

##### 1) Kekontrasan (*Contrast*)

Disebut juga dengan “*sum of squares variance*”. *Contrast* menunjukkan ukuran penyebaran elemen-elemen matriks citra, jika letaknya jauh dari diagonal utama maka nilai kekontrasan semakin besar. Secara visual, nilai kekontrasan adalah variasi antar derajat keabuan suatu daerah citra.

##### 2) *Homogeneity (Invers Difference Moment)*

Tekstur yang homogen adalah mengandung struktur yang berulang dan seragamannya menghasilkan pola tertentu.

##### 3) *Entropy*

Digunakan untuk mengukur ketidak menentuan (*Randomness*) dan akan bernilai 0 untuk citra yang kontras.

##### 4) *Energy*

*Energy* mengacu pada homogenitas dari tekstur, jika nilai energi tinggi maka jumlah area yang homogen besar, dan sebaliknya jika nilai energi rendah maka jumlah area yang homogen sedikit atau kecil.

##### 5) *Korelasi (Correlation)*

Menunjukkan ukuran ketergantungan linear derajat keabuan pixel dan tetangganya dari citra sehingga dapat memberikan petunjuk adanya struktur linear dalam citra (Karyanti, Yuli, 2010).

#### METODE

Metodologi yang digunakan yaitu metode *Euclidean Distance* dengan bantuan aplikasi pemrograman Matlab. Matlab merupakan aplikasi pemrograman yang telah dikenal dalam pembuatan aplikasi penunjang penelitian. Matlab dalam pengenalan pola tekstur menyediakan bermacam tools yang akan mempersingkat waktu penulisan program sehingga peneliti lebih berfokus pada hasil dan inovasi penelitian (Agani, N. 2012).

Pada program pengenalan pola tekstur ini, akan digunakan dua tipe pembelajaran, *supervised learning* (pembelajaran terawasi) dan *unsupervised learning* (pembelajaran tak terawasi). *Image* pola tekstur harus berupa *image grayscale* (hitam putih) sehingga memudahkan untuk proses pengenalan tekstur. Pada ekstraksi ciri objek, akan digunakan tipe *unsupervised learning* dengan salah satu metodenya yaitu metode SPCA dan untuk pengklasifikasian objek digunakan tipe *supervised learning* yaitu fungsi jarak *Euclidean*. Di bawah ini adalah gambaran umum mengenai sistem pengenalan pola tekstur yang akan dibangun.

### 1. Fase Normalisasi

Pada tahap normalisasi / proses awal ini akan dilalui oleh setiap *image*, baik berupa *image* latih (*training*) maupun uji (*testing*). Hal tersebut bertujuan untuk mendapatkan data dengan ukuran yang lebih kecil yang mewakili data *image* asli yang sebenarnya. Pada proses normalisasi ini, resolusi *image* asli akan diubah menjadi ukuran 30x 20 piksel, kemudian diubah ke bentuk matriks berukuran 600 x 1 piksel (satu dimensi).

### 2. Ekstraksi dengan Metode SPCA (*Simple Principal Components Analysis*)

Metode SPCA adalah metode yang biasa digunakan untuk mengurangi tingkat dimensi suatu objek. SPCA juga termasuk dalam pembelajaran yang tak terawasi (*unsupervised learning*) yang diperoleh melalui fitur – fitur normal dari transformasi linear matriks inputan objek. Dengan demikian metode SPCA bertujuan untuk ekstraksi ciri / menajamkan perbedaan - perbedaan pola yang dapat mempermudah dalam pemisahan kategori kelas pada proses klasifikasi.

### 3. Fase Klasifikasi dengan Metode *Euclidean*

Metode *Euclidean* adalah metode pengukuran jarak garis lurus (*straight line*) antara titik X ( $X_1, X_2, \dots, X_n$ ) dan titik Y ( $Y_1, Y_2, \dots, Y_n$ ). Metode *Euclidean* sendiri memiliki rumus (*formula*) pengembangannya sesuai dengan keadaan ruang. Dalam hal ini akan kita gunakan ruang satu dimensi. Jarak satu dimensi dengan titik A(x1) dan B(y1) yang diakuisisi dari data *sample* maupun *testing*.

### 4. Penentuan Keputusan Akhir

Pada fase akhir ini akan diambil keputusan berdasarkan data minimum fungsi jarak *Euclidean* pada fase *testing*. Dengan demikian, dapat diketahui bahwa suatu pola *image* / data baru dapat dikatakan mirip dengan salah satu pola *image training*, melalui fungsi jarak *Euclidean* yang didasarkan pada nilai jarak minimum.

## HASIL DAN PEMBAHASAN



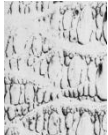
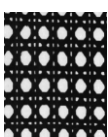
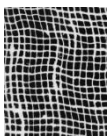
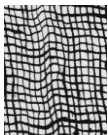

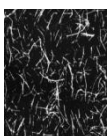
### Database citra tekstur


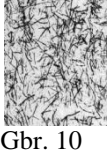
Input untuk pengenalan tekstur ini telah disediakan 10 tekstur Brodatz (Randen, 2011).

a. *Input training* (masukan pembelajaran) terdiri dari 9 buah gambar atau citra tekstur citra *grayscale*.

b. *Input testing* (masukan pengujian) dipilih dari sisa citra *input* yang tidak disertakan (digunakan) pada *inputan training* yaitu gambar 10.

Hasil perhitungan *Normalized Euclidean Distance* menggunakan aplikasi Matlab ditunjukkan pada Tabel 1.

Tekstur	Sandart deviasi	Average	Entropy	Con trast	Corre lation	Energy	Homo geneity	Euclidean Distance
 Gbr. 1	78.1059	172.4224	4.9921	0.9130	0.8989	0.3700	0.8501	14.5319
 Gbr. 2	57.6311	157.1497	5.9411	0.3617	0.9289	0.1475	0.8773	18.1870
 Gbr. 3	48.8750	199.6003	4.2693	1.1907	0.6549	0.5153	0.8489	29.3672
 Gbr. 4	89.8111	77.7800	3.9250	1.0358	0.9110	0.1783	0.8379	99.9903
 Gbr. 5	88.0571	101.1761	5.4262	2.1366	0.8166	0.1191	0.7671	77.0607
 Gbr. 6	87.8826	132.3034	5.1878	1.9526	0.8127	0.2197	0.8085	48.4496
 Gbr. 7	68.1253	146.3339	6.0156	7.2073	0.0246	0.0492	0.4458	28.3046
 Gbr. 8	48.7810	43.1480	4.4963	1.3784	0.5852	0.3141	0.8001	131.9915

 Gbr.9	65.8256	170.8663	5.6259	2.3062	0.6423	0.1935	0.7067	4.0250
 Gbr. 10	63.7068	174.2883	5.5924	2.2978	0.6190	0.1972	0.7052	0

**Script Matlab untuk pengenalan pola citra tekstur adalah sebagai berikut:**

```
% PENGENALAN POLA CITRA TEKSTUR METODE EUCLIDEAN
clear, close all
I1=imread('C:\Users\HP 430\Documents\gbr_1.png');
I2=imread('C:\Users\HP 430\Documents\gbr_2.png');
I3=imread('C:\Users\HP 430\Documents\gbr_3.png');
I4=imread('C:\Users\HP 430\Documents\gbr_4.png');
I5=imread('C:\Users\HP 430\Documents\gbr_5.png');
I6=imread('C:\Users\HP 430\Documents\gbr_6.png');
I7=imread('C:\Users\HP 430\Documents\gbr_7.png');
I8=imread('C:\Users\HP 430\Documents\gbr_8.png');
I9=imread('C:\Users\HP 430\Documents\gbr_9.png');
I10=imread('C:\Users\HP 430\Documents\gbr_line.png');
%Mencari Nilai Rata-Rata, Entropy dan Intensitas Warna Gambar 1
av_1 = mean2(I1);
ent_1 = entropy(I1);
std_1 = std2(I1);
glcm2_1 = graycomatrix(I1);
C1=[std_1, av_1, ent_1]'
%mencari Contrast, Correlation, Energy dan Homogeneity
stats = graycoprops(glcm2_1,{'all'})
%Mencari Nilai Rata-Rata, Entropy dan Intensitas Warna Gambar 2
av_2 = mean2(I2);
ent_2 = entropy(I2);
std_2 = std2(I2);
glcm2_2 = graycomatrix(I2);
C2=[std_2, av_2, ent_2]'
%mencari Contrast, Correlation, Energy dan Homogeneity
stats = graycoprops(glcm2_2,{'all'})
%Mencari Nilai Rata-Rata, Entropy dan Intensitas Warna Gambar 3
av_3 = mean2(I3);
ent_3 = entropy(I3);
std_3 = std2(I3);
glcm2_3 = graycomatrix(I3);
C3=[std_3, av_3, ent_3]'
%mencari Contrast, Correlation, Energy dan Homogeneity
stats = graycoprops(glcm2_3,{'all'})
%Mencari Nilai Rata-Rata, Entropy dan Intensitas Warna Gambar 4
av_4 = mean2(I4);
ent_4 = entropy(I4);
std_4 = std2(I4);
glcm2_4 = graycomatrix(I4);
```

```
C4=[std_4, av_4, ent_4]'  
%mencari Contrast,Correlation,Energy dan Homogeneity  
stats = graycoprops(glcm2_4,{'all'})  
%Mencari Nilai Rata-Rata, Entropy dan Intensitas Warna Gambar 5  
av_5 = mean2(I5);  
ent_5 = entropy(I5);  
std_5 = std2(I5);  
glcm2_5 = graycomatrix(I5);  
C5=[std_5, av_5, ent_5]'  
%mencari Contrast,Correlation,Energy dan Homogeneity  
stats = graycoprops(glcm2_5,{'all'})  
%Mencari Nilai Rata-Rata, Entropy dan Intensitas Warna Gambar 6  
av_6 = mean2(I6);  
ent_6 = entropy(I6);  
std_6 = std2(I6);  
glcm2_6 = graycomatrix(I6);  
C6=[std_6, av_6, ent_6]'  
%mencari Contrast,Correlation,Energy dan Homogeneity  
stats = graycoprops(glcm2_6,{'all'})  
%Mencari Nilai Rata-Rata, Entropy dan Intensitas Warna Gambar 7  
av_7 = mean2(I7);  
ent_7 = entropy(I7);  
std_7 = std2(I7);  
glcm2_7 = graycomatrix(I7);  
C7=[std_7, av_7, ent_7]'  
%mencari Contrast,Correlation,Energy dan Homogeneity  
stats = graycoprops(glcm2_7,{'all'})  
%Mencari Nilai Rata-Rata, Entropy dan Intensitas Warna Gambar 8  
av_8 = mean2(I8);  
ent_8 = entropy(I8);  
std_8 = std2(I8);  
glcm2_8 = graycomatrix(I8);  
C8=[std_8, av_8, ent_8]'  
%mencari Contrast,Correlation,Energy dan Homogeneity  
stats = graycoprops(glcm2_8,{'all'})  
% MENCARI CITRA YANG PALING MIRIP TERHADAP CITRA 10  
dist_1=sqrt((C1(1,1)-C10(1,1))^2+(C1(2,1)-C10(2,1))^2+(C1(3,1)-  
C10(3,1))^2)  
dist_2=sqrt((C2(1,1)-C10(1,1))^2+(C2(2,1)-C10(2,1))^2+(C2(3,1)-  
C10(3,1))^2)  
dist_3=sqrt((C3(1,1)-C10(1,1))^2+(C3(2,1)-C10(2,1))^2+(C3(3,1)-  
C10(3,1))^2)  
dist_4=sqrt((C4(1,1)-C10(1,1))^2+(C4(2,1)-C10(2,1))^2+(C4(3,1)-  
C10(3,1))^2)  
dist_5=sqrt((C5(1,1)-C10(1,1))^2+(C5(2,1)-C10(2,1))^2+(C5(3,1)-  
C10(3,1))^2)  
dist_6=sqrt((C6(1,1)-C10(1,1))^2+(C6(2,1)-C10(2,1))^2+(C6(3,1)-  
C10(3,1))^2)  
dist_7=sqrt((C7(1,1)-C10(1,1))^2+(C7(2,1)-C10(2,1))^2+(C7(3,1)-  
C10(3,1))^2)  
dist_8=sqrt((C8(1,1)-C10(1,1))^2+(C8(2,1)-C10(2,1))^2+(C8(3,1)-  
C10(3,1))^2)  
dist_9=sqrt((C9(1,1)-C10(1,1))^2+(C9(2,1)-C10(2,1))^2+(C9(3,1)-  
C10(3,1))^2)  
dist_10=sqrt((C10(1,1)-C10(1,1))^2+(C10(2,1)-  
C10(2,1))^2+(C10(3,1)-C10(3,1))^2)
```



Input untuk pengenalan tekstur ini telah disediakan 10 tekstur Brodatz (Randen, 2011).

a. *Input training* (masukan pembelajaran) terdiri dari 9 buah gambar atau citra tekstur citra *grayscale*.

b. *Input testing* (masukan pengujian) dipilih dari sisa citra *input* yang tidak disertakan (digunakan) pada *inputan training* yaitu gambar 10.

Keluaran yang dihasilkan dari aplikasi pengenalan tekstur menggunakan fungsi jarak *Euclidean* adalah penilaian fungsi jarak *Euclidean* pada perhitungan kecocokan pola *training* dengan pola *testing* untuk mengenali kemiripan tekstur.

Citra bisa direpresentasikan dengan beberapa cara antara lain dengan energi dan standar deviasi. Dasar pemikiran dari penggunaan energi sebagai fitur untuk pembedaan tekstur adalah bahwasannya distribusi energi dalam domain frekuensi mengidentifikasi sebuah tekstur (Manjunath, 1996).

### Perhitungan Kemiripan Citra tekstur

Salah satu metode yang sering digunakan untuk menghitung kemiripan antara 2 citra digunakan metode *Euclidean Distance*.

*Euclidean Distance* tidak selalu menjadi metode pengukuran kemiripan yang terbaik. Faktanya adalah bahwasanya jarak dalam setiap dimensi dikuadratkan terlebih dahulu sebelum dilakukan penjumlahan. Hal ini memberikan penekanan yang kuat pada fitur-fitur yang memiliki *dissimilarity* (perbedaan) besar. Oleh karena itu, perlu dilakukan normalisasi komponen-komponen fitur tersendiri sebelum menghitung kemiripan citra.

Dari hasil uji coba menunjukkan bahwa :

1. Nilai kontras tertinggi pada gambar 7 yang menunjukkan bahwa letaknya paling jauh dari diagonal utama dan terendah pada gambar 2 yang menunjukkan bahwa letaknya paling dekat dari diagonal utama.
2. Nilai korelasi tertinggi pada gambar 2 dan terendah pada gambar 7 hal ini menunjukkan ukuran ketergantungan linear derajat keabuan pixel dari citra sehingga dapat memberikan petunjuk adanya struktur linear dalam citra.
3. Nilai energi tertinggi pada gambar 3 hal ini menunjukkan bahwa jumlah area yang homogen besar, dan terendah pada gambar 7, hal ini menunjukkan jumlah area yang homogen sedikit atau kecil.
4. Nilai Homogeneity tertinggi pada gambar 2 hal ini menunjukkan teksturnya kurang homogen dan terendah pada gambar 7 hal ini menunjukkan teksturnya paling homogen.

Sebuah citra gambar digital dapat mewakili sebuah matriks yang berukuran M kolom dan N baris. Perpotongan antara kolom dan baris disebut piksel yang merupakan elemen terkecil dari sebuah citra.

Piksel mempunyai dua parameter:

- koordinat
- intensitas (warna)

Nilai yang terdapat pada koordinat  $(x,y)$  adalah  $f(x,y)$  yaitu besar intensitas (warna) dari piksel titik tersebut. Pengenalan pola tekstur citra digital dengan Metode jarak *Euclidean* dengan menggunakan ciri-ciri:

- Intensitas warna ( $\sigma$ )
- Nilai rata-rata ( $\mu$ )

- Entropi (e)
- Energi (E)
- Homogeiniti (H)
- Contrast (C)

Misal sebuah citra  $x$  yang akan diuji, citra mana yang paling mirip dengan citra  $x$ , dengan metode Euclidean dapat ditentukan besarnya jarak antar citra tersebut. Citra yang paling mirip adalah citra yang mempunyai nilai jarak *Euclidean* paling kecil.

Dari hasil perhitungan *Euclidean Distance* menggunakan Matlab dihasilkan :

dist\_1 = 14.5319

dist\_2 = 18.1870

dist\_3 = 29.3672

dist\_4 = 99.9903

dist\_5 = 77.0607

dist\_6 = 48.4496

dist\_7 = 28.3046

dist\_8 = 131.9915

dist\_9 = 4.0250

dist\_10 = 0

Gambar yang paling mirip dengan gambar 10 adalah gambar 9 dengan nilai *Euclidean distance* terkecil yaitu 4.0250 dan gambar yang paling berbeda dengan gambar 10 adalah gambar 8 dengan nilai *Euclidean Distance* terbesar yaitu 131.9915

## PENUTUP

### Kesimpulan

1. Penggunaan *Euclidean Distance* untuk mengetahui kemiripan pola tekstur cukup baik. Hasil identifikasi tekstur tergantung dari nilai intensitas warna ( $\sigma$ ), rata-rata ( $\mu$ ), entropi (e), energi (E), dan homogeiniti (H)
2. *Euclidean Distance* dapat digunakan dalam pendekatan statistik untuk ekstraksi fitur tekstur, dengan menghitung nilai kontras, homogeniti, entropi, energi dan korelasi.
3. Pola tekstur yang paling mirip dengan gambar 10 adalah pola tekstur pada gambar 9 dengan nilai *Euclidean distance* terkecil.

### Saran

Beberapa saran yang bisa diberikan untuk penelitian selanjutnya antara lain :

1. Perlu dilakukan penelitian lanjutan untuk membuat sistem pengenalan pola tekstur secara *realtime*.
2. Perlu dilakukan penelitian lanjutan dengan menambahkan sistem deteksi pola tekstur , sehingga sistem menjadi lebih lengkap dan dapat mengenali berbagai macam pola, selain itu jumlah pola yang digunakan bisa lebih banyak lagi.
3. Perlu dilakukan penelitian lebih lanjut mengenai pengenalan pola tekstur dengan menggunakan metode lain dan hasilnya bisa dibandingkan dengan metode Euclidean distance.

#### **DAFTAR PUSTAKA**

- Agani, Nazori. 2012. **Diktat Kuliah Teknik Komputasi**. Universitas Budi Luhur.
- Alexander, Bunga J. 2009. **Klasifikasi Citra dengan Metode Transformasi Wavelet pada Lima Biji-Bijian**. Skripsi Jurusan Teknik Elektro. Universitas Diponegoro. Semarang
- Al-Tayeche, Rami, And Ahmed, Khalil. 2003. **CBIR: Content Based Image Retrieval. Report Submitted in Partial Fulfillment of the Requirements of Engineering Project Department of Systems and Computer Engineering**. Faculty of Engineering Carleton University.
- Falasev, RS. 2010. **Pengenalan Sidik Jari Manusia dengan Matriks Kookurensi Aras Keabuan (Graylevel Coocurrence Matrix)**. Skripsi Jurusan Teknik Elektro. Universitas Diponegoro. Semarang
- Gonzales, R. C. And P. Wintz. 2004. **Digital Image Processing**. Addison Wesley Pub. Company. USA.
- Gonzales, R. C. And Woods, R. E. 2002. **Digital Image Processing, 2nd ed.**, Prentice Hall, Upper Saddle River, NJ.
- Haralick, R. K. Shanmugam. 1998. **Feature Normalization and Likelihood-Based Similarity Measures for Image Retrieval**. in Pattern Recognition Letters, 22(5):563- 582
- Karyanti, Yuli. 2010. **Pencarian Citra Berdasarkan Filter Lokal Tekstur**. Disertasi Program Doktor Teknologi Informasi. Universitas Gunadarma. Depok
- Kokare, M., Biswas, P., & Chatterji, B. 2007. **Texture Image Retrieval Using Rotated Wavelet Filter**. Pattern Recognition Letters 28 , 1240-1249.
- Manjunath, B., & Ma, W. 1996. **Texture Features for Browsing and Retrieval of Image Data**. IEEE Transactions on Pattern Analysis and Machine Intelligence , 837- 842
- Randen, T. 2011. **Brodatz Textures**. <URL:<http://www.ux.uis.no/~tranden/brodatz.html>>
- Rinaldi, Munir. 2004. **Pengolahan Citra Digital dengan Pendekatan Algoritmik**. Informatika Bandung. Bandung.